ปัญหาที่มักพบเมื่อใช้งานดีเบส

ยืน ภู่วรวรรณ

ดีเบสทรีพลัสเป็นโปรแกรมที่มีการใช้งานกันอย่างแพร่หลายมาก จนเรียกได้ว่าดีเบสทรีพลัสเป็นโปรแกรมที่ทุกคนที่รู้จักกับไมโครคอมพิวเตอร์จะต้องรู้จักและคุ้นเคย ผู้ใช้ดีเบสทรีพลัสบางคนอาจพบปัญหาบางประการ ซึ่งเมื่อค้นหาจากคู่มือดีเบสทรีพลัสแล้วก็ไม่มีบอกไว้และไม่ทราบทางแก้ไข ในฉบับนี้จะประเมินปัญหาต่าง ๆ เท่าที่จะรวบรวมได้และจะชี้แนวทางแก้ไขปัญหาไว้ให้

การใช้คำสั่ง RUN

ดีเบสมีคำสั่ง RUN ที่จะนำไปใช้ในคำสั่ง DOS เช่น คำสั่ง DIR, DEL, COPY ฯลฯ ได้ รูปแบบการใช้คำสั่งจะเป็นดังนี้

.RUN DIR B:

ซึ่งก็เป็นการขอดูไดเรคทอรีในไดรฟ์ B นั่นเอง

แต่บางครั้งเมื่อใช้งานปรากฏว่าใช้ไม่ได้เป็นเพราะอะไร

ประการแรกที่จะต้องทำความเข้าใจคือ คำสั่ง RUN จะออกจากดีเบสโดยการโหลด COMMAND.COM เข้ามาก่อนแล้วจึงทำตามคำสั่งที่ตามมา แต่ถ้าหากใช้หน่วยความจำน้อย เช่น 256KB ทำให้เหลือหน่วยความจำไม่มากพอ เพราะมีการใช้ดีเบสอยู่ ทำให้ไม่สามารถใช้คำสั่งนี้ได้

เหตุผลประการต่อมาคือ เมื่อดีเบสโหลด COMMAND.COM ปรากฏว่า ณ ที่ดีฟอลต์ไดรฟ์ไม่มีโปรแกรม COMMAND.COM อยู่ ทำให้ RUN ไม่ได้ หรือกรณีใช้ฮาร์ดดิสค์โปรแกรมดีเบสอยู่ในไดเรคทอรีย่อย ทำให้มองไม่เห็นโปรแกรม COMMAND.COM จึงไม่สามารถโหลดได้

ทางแก้ไขทำได้โดยการสร้าง PATH ให้สามารถเรียก COMMAND.COM ในไดรฟ์ที่ต้องการได้ เราอาจกำหนด PATH ก่อนการเรียกดีเบส โดยใช้คำสั่งในดอสดังนี้

A > PATH = B:

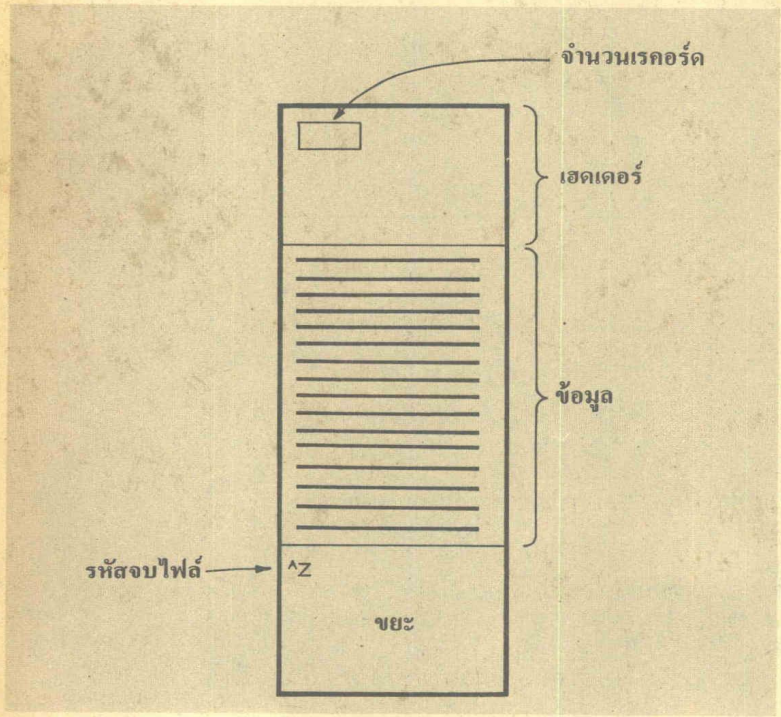
หรือ

C > PATH = C:

หรือจะใช้การกำหนดใน COMSPEC ก็ได้โดยใช้

C> SET COMSPEC = C:
\COMMAND.COM

อย่างไรก็ตาม การใช้ DOS เวอร์ชัน 3.X อาจไม่พบปัญหาคำสั่ง RUN เพราะ DOS 3.X ใส่ COMMAND.COM ไว้ถาวรแล้ว



รูปที่ 1 โครงสร้างข้อมูล .dbf

ปัญหาเกี่ยวกับ dbf ไฟล์

ในการใช้ไฟล์ .dbf เก็บข้อมูลเมื่อใช้งานอยู่บางครั้งอาจพบปัญหาบางอย่าง ซึ่งปัญหาเหล่านี้มีหลายอย่างที่จะต้องทำความเข้าใจและหาทางแก้ไขให้ถูกจุด

โครงสร้างของแฟ้ม dbf ประกอบด้วยข้อมูลดังรูปที่ 1 และโครงสร้างของเฮดเดอร์ประกอบด้วยโครงสร้างดังตารางที่ 1

ไบต์ที่	จำนวน	ความหมาย
0	1 ไบต์	รหัส 03 หมายถึง dbf
1–3	3 ไบต์	วันที่
4–7	32 บิต	เลขจำนวนเรคอร์ด
8–9	16 บิต	จำนวนของไบต์ในเฮดเดอร์
10–11	16 บิต	จำนวนไบต์ในเรคอร์ด
12–14	3 ไบต์	สงวนไว้
15–27	13 ไบต์	สงวนไว้สำหรับงาน multi user
28–31	20 ไบต์	สงวนไว้
32–n	กลุ่มละ 32 ไบต์	รายละเอียดบอกฟิลด์
n+1	1 ไบต์	0D เป็นรหัสแบ่งแยกฟิลด์

ตารางที่ 1 โครงสร้างของเฮดเดอร์ไฟล์ .dbf

2

```
. list stru
Structure for database: C:dm1.dbf
Number of data records:       3
Date of last update    : 09/05/87
Field  Field Name  Type       Width     Dec
    1  IDNO        Character      4
    2  NAME        Character     15
    3  ONHAND      Numeric        4
    4  COST        Numeric       10       2
** Total **                      34

. list
Record#  IDNO NAME            ONHAND        COST
      1  A001 DBASE III+          44      180.00
      2  A002 LOTUS 123           18      255.00
      3  A003 MULTIPLAN           15      321.00
```

3

```
. ! debug
-ndm1.dbf
-l
-d
4A5E:0100  03 57 09 05 03 00 00 00-A1 00 22 00 00 00 00 00   .W........".....
4A5E:0110  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
4A5E:0120  49 44 4E 4F 00 00 00 00-00 00 00 43 0F 00 8E 69   IDNO.......C...i
4A5E:0130  04 00 00 00 01 00 00 00-00 00 00 00 00 00 00 00   ................
4A5E:0140  4E 41 4D 45 00 00 00 00-00 00 00 43 13 00 8E 69   NAME.......C...i
4A5E:0150  0F 00 00 00 01 00 00 00-00 00 00 00 00 00 00 00   ................
4A5E:0160  4F 4E 48 41 4E 44 00 00-00 00 00 4E 22 00 8E 69   ONHAND.....N"..i
4A5E:0170  04 00 00 00 01 00 00 00-00 00 00 00 00 00 00 00   ................

-d
4A5E:0180  43 4F 53 54 00 00 00 00-00 00 00 4E 26 00 8E 69   COST.......N&..i
4A5E:0190  0A 02 00 00 01 00 00 00-00 00 00 00 00 00 00 00   ................
4A5E:01A0  0D 20 41 30 30 31 44 42-41 53 45 20 49 49 49 2B   . A001DBASE III+
4A5E:01B0  20 20 20 20 20 20 20 34-34 20 20 20 20 31 38 30          44    180
4A5E:01C0  2E 30 30 20 41 30 30 32-4C 4F 54 55 53 20 31 32   .00 A002LOTUS 12
4A5E:01D0  33 20 20 20 20 20 20 20-20 31 38 20 20 20 20 32   3         18    2
4A5E:01E0  35 35 2E 30 30 20 41 30-30 33 4D 55 4C 54 49 50   55.00 A003MULTIP
4A5E:01F0  4C 41 4E 20 20 20 20 20-20 20 20 31 35 20 20 20   LAN          15

-d
4A5E:0200  20 33 32 31 2E 30 30 20-41 30 30 35 57 4F 52 44    321.00 A005WORD
4A5E:0210  53 54 41 52 20 20 20 20-20 20 20 20 20 34 37 20   STAR           47
4A5E:0220  20 20 20 31 34 30 2E 30-30 20 41 30 30 34 53 55      140.00 A004SU
4A5E:0230  50 45 52 43 41 4C 43 20-20 20 20 20 20 20 20 33   PERCALC         3
4A5E:0240  35 20 20 20 20 31 38 30-2E 30 30 1A 72 65 63 74   5    180.00.rect
4A5E:0250  6F 72 79 2C 0D 0A 6F 72-20 64 69 72 65 63 74 6F   ory,..or directo
4A5E:0260  72 79 20 6E 6F 74 20 65-6D 70 74 79 0D 0A 00 38   ry not empty...8
4A5E:0270  38 4D 75 73 74 20 73 70-65 63 69 66 79 20 4F 4E   8Must specify ON
```

รหัสบอกดีเบสไฟล์ *วันที่* *จำนวนเรคอร์ด 3* *รหัสจบไฟล์*

4

```
-ecs:104
4A5E:0104  03.05
-dcs:100
4A5E:0100  03 57 09 05 05 00 00 00-A1 00 22 00 00 00 00 00   .W........".....
4A5E:0110  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
4A5E:0120  49 44 4E 4F 00 00 00 00-00 00 00 43 0F 00 8E 69   IDNO.......C...i
4A5E:0130  04 00 00 00 01 00 00 00-00 00 00 00 00 00 00 00   ................
4A5E:0140  4E 41 4D 45 00 00 00 00-00 00 00 43 13 00 8E 69   NAME.......C...i
4A5E:0150  0F 00 00 00 01 00 00 00-00 00 00 00 00 00 00 00   ................
4A5E:0160  4F 4E 48 41 4E 44 00 00-00 00 00 4E 22 00 8E 69   ONHAND.....N"..i
4A5E:0170  04 00 00 00 01 00 00 00-00 00 00 00 00 00 00 00   ................
-
```

กรณีที่เมื่อใช้คำสั่ง LIST แล้วมีจำนวนไม่เท่ากับที่บอกไว้ในคำสั่ง LIST STRU

หลายท่านคงประสบปัญหาโดยเฉพาะอย่างยิ่งขณะใช้งานดีเบสอยู่แล้วเกิดมีปัญหาทางด้านฮาร์ดแวร์หรือไฟดับหรือเครื่องแฮง ทำให้ข้อมูลในดีเบสเกิดมีปัญหา ซึ่งเมื่อ list แล้วข้อมูลกับจำนวนเรคอร์ดไม่เท่ากัน เช่น เมื่อเราป้อนข้อมูลไปแล้ว 200 เรคอร์ด แต่ที่หัวไฟล์บอกว่ามี 100 เรคอร์ด ดังนั้น เราจะแสดงหรือใช้ข้อมูลได้เพียง 100 เรคอร์ด และที่เหลือจะใช้ไม่ได้

หรือบางทีที่เฮดเดอร์บอกจำนวนเรคอร์ด 300 เรคอร์ด แต่ข้อมูลมี 200 เรคอร์ด ดังนั้นเมื่อ list มาถึงเรคอร์ด 200 จะปรากฏว่าพบรหัส end of file ในที่ไม่พึงปรารถนา

เราสามารถแก้ไขปัญหานี้ด้วย DEBUG ลองดูปัญหาเมื่อ list ไฟล์และ list structure แล้วไม่ตรงกัน เช่น ไฟล์ DM1 มี 5 record แต่ list เห็นแค่ 3 ดังรูปที่ 2

เมื่อใช้ดีบักดูจะเห็นว่าเรามี 5 เรคอร์ด เพราะ 1A รหัส EOF อยู่ที่เรคอร์ดสุดท้าย ดังรูปที่ 3

5

```
. use dm1
. list stru
Structure for database: C:dm1.dbf
Number of data records:       5
Date of last update    : 09/05/87
Field  Field Name  Type        Width     Dec
    1  IDNO        Character       4
    2  NAME        Character      15
    3  ONHAND      Numeric         4
    4  COST        Numeric        10       2
** Total **                      34

. list
Record#  IDNO NAME                ONHAND        COST
      1  A001 DBASE III+              44      180.00
      2  A002 LOTUS 123               18      255.00
      3  A003 MULTIPLAN               15      321.00
      4  A005 WORDSTAR                47      140.00
      5  A004 SUPERCALC               35      180.00
```

จากการขอดูข้อมูลด้วยดีบักพบว่ามีจำนวนข้อมูล 5 เรคอร์ด แต่ที่เขียนไว้ที่เฮดเดอร์มีเพียง 3 เรคอร์ด ซึ่งเราต้องแก้ไขตัวเลขให้เป็น 5 เรคอร์ด ตามข้อมูลที่ป้อนไว้โดยการใช้คำสั่ง E ในดีบัก ดังรูปที่ 4 ปิดท้ายด้วยคำสั่ง W. เพื่อเขียนที่แก้ไขทั้งหมดลงไฟล์ ดังนี้

```
-w
Writing 014C bytes
-q
```

ออกจากดีบักกลับเข้าสู่ดีเบสใหม่ คราวนี้เมื่อ list structure หรือ list จะได้ข้อมูลครบตามที่ต้องการ ดังรูปที่ 5

เมื่อใช้งานไฟล์ปรากฏข้อความ "Too many files open"

เอ็มเอสดอสจะกันพื้นที่บัฟเฟอร์และจำนวนไฟล์สำหรับการเปิดไฟล์สำหรับใช้งาน โดยปกติ MSDOS จะเปิดไฟล์มาตรฐาน 5 ไฟล์ไว้แล้วคือ ไฟล์จอภาพ ไฟล์คีย์บอร์ด ไฟล์ข้อผิดพลาดมาตรฐาน ไฟล์ AUX และไฟล์ PRN โดยปกติเอ็มเอสดอสจะติดตั้งระบบโดยใช้ CONFIG.SYS เพื่อบอกจำนวนไฟล์ โดยใช้คำสั่งกำหนดจำนวนไฟล์และบัฟเฟอร์ โดยเขียนข้อความลงไปว่า

FILES = 20
BUFFERS = 10

หากไม่มีการกำหนดใน CONFIG.SYS เอ็มเอสดอสจะยอมให้เปิดไฟล์ได้อีกประมาณ 2–4 ไฟล์ ดังนั้นจึงปรากฏข้อความ "Too many files open"

ข้อผิดพลาด "File already open"

การเปิดไฟล์ทุกครั้งดีเบสจะสร้างบัฟเฟอร์ข้อมูลไว้ และมีสภาพบอกการเปิดไฟล์ไว้ เช่น บางครั้งใช้งานอยู่แล้วเราจะเรียกไฟล์ข้อมูลมา ปรากฏว่าเกิดข้อความ file already open อาจเกิดจากปัญหาหลายอย่าง ดังนี้

การเปิดไฟล์ไว้ในพื้นที่อื่น เช่น SELECT 5 แล้วเปิดไว้จนลืม และเมื่อเรามา SELECT ใหม่แล้วจะเปิดไฟล์เดิมจะปรากฏข้อความดังกล่าว ดังนั้นต้องปิดไฟล์ก่อนโดยใช้คำสั่ง CLOSE DATABASE ก่อน

บางครั้งเมื่อใช้คำสั่งสร้างรายงาน เช่น create report จะมีการเปิดไฟล์ข้อมูลบางไฟล์ หรือบางครั้งมีการเปิดไฟล์ที่เชื่อมโยงกับงานบางอย่าง ทำให้สถานะการเปิดไฟล์ค้างไว้ ทำให้ไม่สามารถเปิดไฟล์ใหม่ได้ เราอาจทำได้ด้วยการใช้คำสั่ง CLEAR ALL หรือ CANCEL ดู ถ้ายังไม่ได้ผลก็คงต้อง QUIT แล้วกลับเข้ามาใหม่

ปัญหาการใช้ INDEX

ในกรณีที่ต้องใช้ไฟล์ INDEX อาจพบปัญหาหลายอย่างที่ยากต่อการเข้าใจได้ และบางครั้งก็ยากต่อการแก้ไข เราลองมาทำความเข้าใจกับ INDEX แบบพื้นฐานกันก่อน ในรูปที่ 6

ดีเบสให้เราสร้าง .NDX ไฟล์ที่สัมพันธ์กับ .DBF ไฟล์ในฟิลด์ต่าง ๆ กันได้ 7 ไฟล์ ดังนั้น เมื่อใช้งานจะต้องคำนึงถึงอินเด็กซ์ทุก ๆ ไฟล์ที่เชื่อมกับ .DBF ตัวเดียวกันด้วย

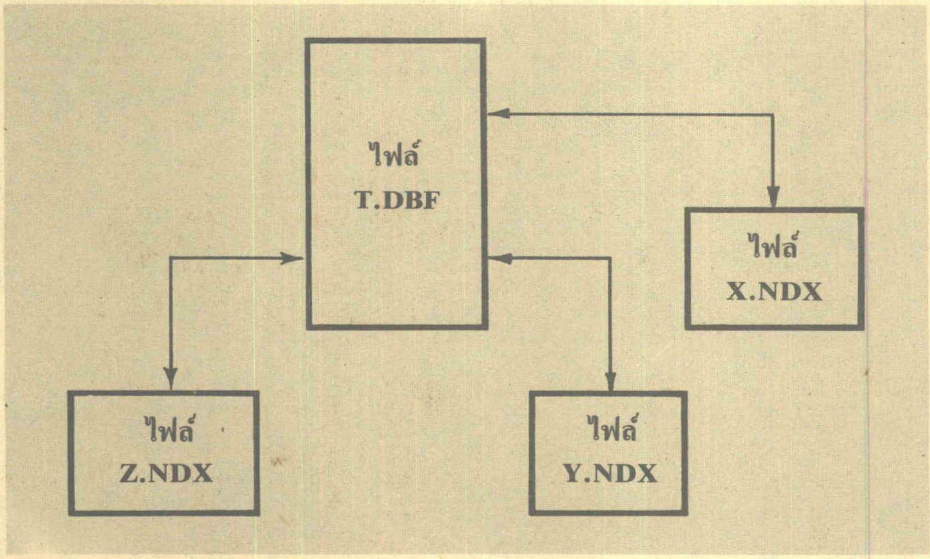
กรณีที่ต้องมีการแก้ไข เพิ่มเติมข้อมูล เราต้องเปิดไฟล์ให้ครบ เช่น

.USE T INDEX X,Y,Z

หรือจะใช้

.USE T
.SET INDEX TO X,Y,Z

หลังจากนั้นจึงใช้คำสั่ง APPEND หรือ EDIT ได้โดยไม่ทำให้ข้อมูลอื่นมีปัญหา



รูปที่ 6 โครงสร้างการเชื่อมของ .NDX กับ .DBF

ปัญหาที่เกิดกับอินเด็กซ์ไฟล์มีหลายประการ ที่พบกันเช่น เกิดข้อความ "record out of range" กรณีเกิดจากการป้อนหรือแก้ไขข้อมูลใน .DBF ไฟล์ แต่ไม่มีการไปปรับปรุง .NDX กรณีนี้จะต้องใช้คำสั่ง REINDEX ตามมา

กรณีเมื่อไฟล์มีขนาดใหญ่มาก

ระบบที่ใช้งานจริงที่ประกอบด้วยไฟล์หลาย ๆ ไฟล์รวมกันเป็นฐานข้อมูล และไฟล์ที่ใช้เมื่อมี .NDX ด้วยและมีขนาดใหญ่ควรหลีกเลี่ยงคำสั่ง DELE หรือการลบเรคอร์ดเพราะการลบข้อมูลจะมีปัญหาในการปรับปรุงไฟล์มาก โดยปกติเราจะใช้การทำเครื่องหมายเป็นฟิลด์ใดฟิลด์หนึ่งเพื่อบอกว่าเรคอร์ดนี้เราไม่ต้องการ แต่ให้ข้อมูลอยู่อย่างเดิม และเมื่อต้องการเพิ่มข้อมูลจะทำการค้นหาเรคอร์ดเหล่านี้แล้วแทนข้อมูลใหม่เข้าไปแทน

ส่งท้ายกับการลบไฟล์โดยไม่ตั้งใจ

หากลบไฟล์โดยไม่ตั้งใจจะทำอย่างไรดี กรณีนี้อย่าเพิ่งตกใจ เรามีวิธีการกู้คืนได้ ขั้นแรกเราต้องไม่มีการสร้างไฟล์ใด ๆ ขึ้นใหม่โดยเด็ดขาด จากนั้นให้หาโปรแกรมยูทิลิตี้บางโปรแกรมที่จะช่วยได้ เช่น NU (Norton Utility) ซึ่งจะมียูทิลิตี้ในการแก้ไขไฟล์ที่ลบแล้วได้ แต่ต้องมีขบวนการหลายอย่างพอควร ขอให้ศึกษาและทดลองทำดูจะเห็นประโยชน์อย่างแท้จริง สำหรับในฉบับนี้จะไม่กล่าวถึงรายละเอียดขั้นตอนทั้งหมด ขอให้อ่านในมุมโอเอสในฉบับที่แล้วครับ